NAS PARALLEL BENCHMARK RESULTS 3-94

David H. Bailey, Eric Barszcz, Leonardo Dagum and Horst D. Simon[1]

RNR Technical Report RNR-94-006
March 21, 1994

Abstract

The NAS Parallel Benchmarks have been developed at NASA Ames Research Center to study the performance of parallel supercomputers. The eight benchmark problems are specified in a "pencil and paper" fashion. In other words, the complete details of the problem to be solved are given in a technical document, and except for a few restrictions, benchmarkers are mostly free to select the language constructs and implementation techniques best suited for a particular system.

This paper presents performance results of various systems using the NAS Parallel Benchmarks. These results represent the best results that have been reported to us for the specific systems listed. Some changes and clarifications to the benchmark rules are also described.

[1] Bailey and Barszcz are employees of NASA Ames Research Center. Dagum and Simon are employees of Computer Science Corp., and their work was funded by the NASA Ames Research Center under contract NAS2-12961. Mailing address for all authors: NASA Ames Research Center, Mail Stop T27A-1, Moffett Field, CA 94035-1000. This report is available electronically on the WWW at URL address: http://www.nas.nasa.gov/RNR/Parallel/NPB/results_3_94.html or by sending an email request to leo@nas.nasa.gov

1 Introduction

The Numerical Aerodynamic Simulation (NAS) Program, located at NASA Ames Research Center, is dedicated to advancing the science of computational aerodynamics. One key goal of the NAS organization is to demonstrate by the year 2000 an operational computing system capable of simulating an entire aerospace vehicle system within a computing time of one to several hours. It is currently projected that the solution of this grand challenge problem will require a computer system that can perform scientific computations at a sustained rate approximately one thousand times faster than 1990 generation supercomputers. Most likely such a computer system will employ hundreds or even thousands of processors operating in parallel.

In order to objectively measure the performance of various highly parallel computer systems and to compare them with conventional supercomputers, we along with other scientists in our organization have devised the NAS Parallel Benchmarks (NPB). Note that the NPB are distinct from the High Speed Processor (HSP) benchmarks and procurements. The HSP benchmarks are used for evaluating production supercomputers for procurement, whereas the NPB are for studying massively parallel processor (MPP) systems not necessarily tied to a procurement.

The NPB are a set of eight benchmark problems, each of which focuses on some important aspect of highly parallel supercomputing for aerophysics applications. Some extension of Fortran or C is required for implementations, and reasonable limits are placed on the usage of assembly code and the like, but otherwise programmers are free to utilize language constructs that give the best performance possible on the particular system being studied. The choice of data structures, processor allocation and memory usage are generally left open to the discretion of the implementer.

The eight problems consist of five "kernels" and three "simulated computational fluid dynamics (CFD) applications". Each of these is defined fully in [2]. The five kernels are relatively compact problems, each emphasizing a particular type of numerical computation. Compared with the simulated CFD applications, they can be implemented fairly readily and provide insight as to the general levels of performance that can be expected on these specific types of numerical computations.

The simulated CFD applications, on the other hand, usually require more effort to implement, but they are more indicative of the types of actual data movement and computation required in state-of-the-art CFD application codes. For example, in an isolated kernel a certain data structure may be very efficient on a certain system, and yet this data structure would be inappropriate if incorporated into a larger application. By comparison, the simulated CFD applications require data structures and implementation techniques that are more typical of real CFD applications.

Space does not permit a complete description of these benchmark problems. A more detailed description of these benchmarks, together with the rules and restrictions associated with the benchmarks, may be found in [1]. The full specification of the benchmarks is given in [2].

Sample Fortran programs implementing the NPB on a single processor system are available as an aid to implementors. These programs, as well as the benchmark document itself, are available from the following address: NAS Systems Division, Mail Stop 258-6, NASA Ames Research Center, Moffett Field, CA 94035, attn: NAS Parallel Benchmark Codes or by sending an email request to: bm-codes@nas.nasa.gov. The sample codes are provided on Macintosh floppy disks and contain the Fortran source codes, "README" files, input data files, and reference output data files for correct implementations of the benchmark problems. These codes have been validated on a number of computer systems ranging from conventional workstations to supercomputers.

There are now two standard sizes for the NAS Parallel Benchmarks; these will be referred to as the Class A and Class B size problems. The nominal benchmark sizes for the Class A and Class B are listed in Tables 1a and 1b respectively. These tables also give the standard floating point operation (flop) counts for the two classes of problems. Note that in the case of MG the grid size is unchanged, but a greater flop count results from changes in the inner loop iterations. We insist that those wishing to compute performance rates in millions of floating point operations per second (Mflop/s) use these standard flop counts. The tables contain Mflop/s rates calculated in this manner for the current fastest implementation on one processor of the Cray Y-MP for Class A and one processor of the Cray C90 for Class B. Note, however, that in Tables 2 through 9, performance rates are *not* cited in Mflop/s; we present instead the actual run times (and, equivalently, the performance ratios). We suggest that these, and not Mflop/s, be examined when comparing different systems and implementations.

The current best Cray C90 result for EP employs a table lookup scheme that will be disallowed in the future (see Section 2 below), for this reason no Mflop/s rate appears for EP in table 1b.

Benchmark Name	Abbreviation	Nominal Size	Operation Count ($\times 10^9$)	Mflop/s on Y-MP/1
Embarrassingly Parallel	EP	2^{28}	26.68	211
Multigrid	MG	256^3	3.905	176
Conjugate Gradient	CG	$14,000$	1.508	127
3-D FFT PDE	FT	$256^2 \times 128$	5.631	196
Integer Sort	IS	$2^{23} \times 2^{19}$	0.7812	68
LU Simulated CFD Application	LU	64^3	64.57	194
SP Simulated CFD Application	SP	64^3	102.0	216
BT Simulated CFD Application	BT	64^3	181.3	229

Table 1a: Standard Operation Counts and YMP/1 Mflop/s for Class A Size Problems

Benchmark Name	Abbreviation	Nominal Size	Operation Count ($\times 10^9$)	Mflop/s on C90
Embarrassingly Parallel	EP	2^{30}	100.9	na
Multigrid	MG	256^3	18.81	498
Conjugate Gradient	CG	$75,000$	54.89	447
3-D FFT PDE	FT	512×256^2	71.37	560
Integer Sort	IS	$2^{25} \times 2^{21}$	3.150	244
LU Simulated CFD Application	LU	102^3	319.6	493
SP Simulated CFD Application	SP	102^3	447.1	627
BT Simulated CFD Application	BT	102^3	721.5	572

Table 1b: Standard Operation Counts and C90 Mflop/s for Class B Size Problems

In the following, each of the eight benchmarks will be briefly described, and then the best performance results we have received to date for each computer system will be given in Tables 2 through 9. These tables include run times and performance ratios. The performance ratios compare individual timings with the current best time on that benchmark achieved on one processor of either a Cray Y-MP (for Class A) or a Cray C90 (for Class B). The run times in each case are elapsed time of day figures, measured in accordance with the specifications given in [2].

With the exception of the Integer Sort benchmark, these standard flop counts were determined by using the hardware performance monitor on either the Cray Y-MP or the Cray C90, and we believe that they are close to the minimal counts required for these problems. In the case of the Integer Sort benchmark, which does not involve floating-point operations, we selected a value approximately equal to the number of integer operations required, in order to permit the computation of performance rates analogous to Mflop/s rates. We reserve the right to change these standard flop counts in the future if deemed necessary.

The NAS organization reserves the right to verify any NPB results that are submitted to us. We may, for example, attempt to run the submitter's code on another system of the same configuration as that used by the submitter. In those instances where we are unable to reproduce the submitter's supplied results (allowing a 5% tolerance) our policy is to alert the submitter of the discrepancy and allow him or her until the next release of this report to resolve the discrepancy. If the discrepancy is not resolved to our satisfaction, then our own observed results, and not the submitter's results, will be reported. This policy will apply to all results we receive and publish.

Whenever possible, we have tried to credit the actual individuals and organizations who have contributed the performance results cited in the tables. In these citations, NAS denotes the NAS Applied Research Branch at NASA Ames (including both NASA civil servants and Computer Science Corp. contractors); RIACS denotes the parallel systems division of the Research Institute for Advanced Computer Science, which is located at NASA Ames; BBN denotes Bolt, Beranek and Newman; BCS denotes Boeing Computer Services; CRI denotes Cray Research, Inc.; Fujitsu denotes Fujitsu America, Inc.; KSR denotes Kendall Square Research Corp.; IBM denotes International Business Machines, Inc.; Intel denotes the Supercomputer Systems Division of Intel Corp.; MasPar denotes MasPar Computer Corp.; Meiko denotes Meiko Scientific

Corp.; and TMC denotes Thinking Machines, Inc. Where no individual citation is made for a specific model, the results are due to vendor staff.

This paper reports benchmark results on the following systems: TC2000 by Bolt, Beranek and Newman (BBN); YMP, EL, C90, and T3D by Cray Research Inc. (CRI); Paragon and iPSC/860 by Intel; SP-1 and RS6000-590 by International Business Machines (IBM); VPP500 by Fujitsu; KSR1 and KSR2 by Kendall Square Research; ADENART by Kyoto University and Matsushita Electric Industrial Co.; MP-1 and MP-2 by MasPar Computer Corp.; CM-2, CM-200, CM-5, and CM-5E by Thinking Machines Corp. (TMC); CS-1 by Meiko Scientific; nCUBE-2S by nCUBE; and clusters of distributed workstations including Sparcstation's by Sun; RS6000's by IBM; and 4D25's by SGI. Entries in the tables are ordered alphabetically by vendor, except for distributed workstation results which appear last.

Unfortunately, the limited space in this report does not permit discussion of the methods used in any of these implementations. However, references to technical papers describing these methods have been included whenever such papers are available. In particular, details of the implementation of these benchmarks on the TC2000, the CM2, the CM200, the SP-1 and the IBM Cluster may be found in [4, 5, 9, 11]. General discussion on architectural requirements for the benchmarks may be found in [6]. Readers are referred to these documents for full details.

This report includes a number of new results including previously unpublished Cray C90 and T3D results, Fujitsu VPP500 results, Intel Paragon (with OSF1.2 and with SunMos) results, IBM RS6000-590 and SP-1 results, Kyoto/Matsushita ADENART results, Kendall Square KSR1 and KSR2 results, nCUBE-2S results, and Thinking Machines CM-5E results. Some results using Parasoft Express on distributed workstations which have not previously appeared in this report are also included (results with PVM, p4 and Linda on this same cluster may be found in [11]). These results are listed under "Parasoft IBM (token)", used a cluster of nine IBM RS6000-320H workstations with 25 MHz clock rate, 16 MB memory and a token ring interconnect capable of 16 MBits/sec transfer rates. All the Class B results are new, and many improved Class A results are also presented. These improvements reflect improvements both in compilers and implementations. Efforts are currently underway to port the NAS Parallel Benchmarks on other systems, and we hope to have more results in the future.

2 Benchmark Changes

Because the benchmarks are specified in only a pencil and paper fashion, it is inevitable that loopholes develop whereby the benchmark rules are not violated but the benchmark intent is defeated. This section addresses changes to be made in the Embarrassingly Parallel (EP) and Conjugate Gradient (CG) benchmark specification in order to close some loopholes that have developed with these kernels.

Eventually we hope that parallel computing technology will advance to the point where we will be able to measure performance by providing source code, rather than pencil and paper, benchmark descriptions. However, the current lack of a common parallel language or architectural paradigm prohibits our movement in this direction. In the interim, we intend to make public the benchmark implementations submitted to us by the vendors. There will be a 6 month delay between receipt of an implementation and its public disclosure. It is now a condition of benchmark result submission that NASA be allowed to disclose the source code 6 months after the submission.

2.1 Changes to EP

The intent of the EP benchmark is to provide an accuracy and performance check on the Fortran LOG and SQRT intrinsics and to act as an easy kernel which vendors can readily implement on prototype systems. There are two possible loopholes in its implementation which are here disallowed. Results employing these loopholes will not be reported in future releases of this report.

The first loophole involves using a table lookup scheme to compute the SQRT and LOG functions used to generate Gaussian pseudorandom numbers. When the resulting numbers are close to the histogram boundaries in the verification test, a full precision evaluation of these intrinsics is employed. Thus the scheme passes all the verification tests yet defeats the intent of this benchmark.

The second loophole involves replacing calls to the SQRT and LOG intrinsics by a single call to a Fortran coded function that returns the SQRT(-LOG(X)). Again this scheme will pass the verification test yet does not satisfy the intent since the Fortran intrinsic functions have not been employed in the implementation.

Two changes are here made to the benchmark specification. First, two checksums are now required as part of the verification test. Second, only Fortran intrinsic functions (or equivalent calls to the standard C math library) may be used for SQRT and LOG.

2.2 Changes to CG

The intent of the CG benchmark is to test the performance of the system for unstructured grid computations which by their nature require irregular long distance communication or memory access. The benchmark essentially requires computing a sparse matrix-vector product. Rather than distribute a multi-Mbyte file for the matrix, the compact subroutine **makea** is supplied to generate a random sparse matrix. The **makea** procedure generates a sparse matrix by summing outer products of random sparse vectors. This construction is intended to preclude the clever use of *a priori* knowledge of the matrix structure to reduce the communication requirement.

Nonetheless, by saving the random vectors used in **makea**, it is possible to reformulate the sparse matrix-vector multiply and its associated irregular communication in a way such that communication is substantially reduced, and only a few dense vectors are communicated. All sparse operations can be kept local to the processing nodes.

Although this scheme of matrix-vector multiplication may be considered to satisfy the the rules of the CG benchmark, it defeats its intended purpose of measuring random communication performance. Therefore this scheme is no longer allowed and results employing this loophole will not be reported in future releases of this report. A strict interpretation of the benchmark specification [2] precludes this scheme since it is clearly stated that the conjugate gradient method will be used to compute the solution z to $Az = x$, and as part of this method the vector q must be computed via the product $q = Ap$. This means the matrix A must be used, not the vectors employed in its construction.

3 Kernel Results

3.1 Embarrassingly Parallel (EP) Benchmark

The first of the five kernel benchmarks is an "embarrassingly parallel" problem. In this benchmark, two-dimensional statistics are accumulated from a large number of Gaussian pseudorandom numbers, which are generated according to a particular scheme that is well-suited for parallel computation. This problem is typical of many "Monte-Carlo" applications. Since it requires almost no communication, in some sense this benchmark provides an estimate of the upper achievable limits for floating point performance on a particular system.

Results for the embarrassingly parallel benchmark are shown in Table 2. Not all systems exhibit high rates on this problem. This appears to stem from the fact that this benchmark requires references to several mathematical intrinsic functions, such as the Fortran routines AINT, SQRT, and LOG, and evidently these functions are not highly optimized on some systems.

Results which have employed the reduced precision table lookup scheme described in Section 2.1 are marked by an asterisk. Since the Cray C90 results were computed in this manner, performance ratio's for the Class B size are provided only for systems also employing the table lookup scheme. The SunMos-turbo operating system for the Paragon allows both i860 processors on the node to be used for computation (in regular SunMos and OSF the second processor is used purely for communication).

Intel iPSC/860 and Paragon results are due to J. Baugh of Intel. CM-2, CM-200 and CM-5 results are due to J. Richardson of TMC. KSR1 and KSR2 results are due to S. Breit (KSR), J. Singer (U. Houston), and G. Shah (Georgia Tech). VPP500 results are due to B. Elton of Fujitsu. Distributed workstation results are due to S. White of Emory University [12] except for the SGI results which are due to D. Browning of the NAS System Development branch. The "Mixed-A" computer system consisted of 16 Sun Sparc 1's, one Sun IPC, one Sun Sparc2, 11 Sun SLC's, three IBM RS6000 model 550's, one IBM RS6000 model 530, and one NeXT machine. The listed PVM results used PVM 2.4 and Ethernet.

3.2 Multigrid (MG) Benchmark

The second kernel benchmark is a simplified multigrid kernel, which solves a 3-D Poisson PDE. This problem is simplified in the sense that it has constant rather than variable coefficients as in a more realistic application.

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
BBN TC2000	Dec 91	64	284.0	0.44
Cray Y-MP	Aug 92	1	126.2	1.00
		8	15.9	7.95
Cray Y-MP EL	Feb 94	1	228.7*	0.6*
		4	58.8*	2.1*
		8	30.9*	4.1*
Cray C-90	Feb 94	1	23.8*	5.3*
		4	6.0*	21.0*
		16	1.6*	81.4*
Cray T3D	Feb 94	32	10.69*	11.8*
		64	5.36*	23.5*
		128	2.67*	47.3*
		256	1.34*	94.2*
Fujitsu VPP500	Feb 94	1	48.45	2.6
		4	12.36	10.2
		16	3.17	39.8
IBM RS6000-590	Feb 94	1	68.1*	1.9*
IBM SP-1	Feb 94	8	13.77*	9.2*
		16	6.92*	18.2*
		32	3.48*	36.3*
		64	1.72*	73.4*
Intel iPSC/860	May 92	32	102.7	1.23
		64	51.4	2.46
		128	25.7	4.91
Intel Paragon (OSF1.2)	Mar 94	64	10.45	12.1
		128	5.24	24.1
		256	2.66	47.4
		512	1.38	91.4
Intel Paragon (SunMos turbo)	Mar 94	64	5.27	23.9
		128	2.76	45.7
		256	1.46	86.4
Kendall Square KSR1	Oct 93	16	101.9	1.2
		32	51.4	2.5
		64	26.0	4.9
		128	12.8	9.9
Kendall Square KSR2	Feb 94	32	24.8	5.1
Kyoto/Matsushita ADENART	Feb 94	256	32.9	3.8
MasPar MP-1	Aug 92	4K	248.0	0.51
		16K	69.3	1.82
MasPar MP-2	Nov 92	16K	22.4	5.63
Meiko CS-1	Aug 92	16	116.8	1.08

Table 2a: Results of the Class A Embarrassingly Parallel (EP) Benchmark (* indicates result based on reduced precision table lookup) (cont'd)

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
nCUBE-2S	Mar 94	64	83.8	1.51
		128	41.93	3.01
		256	20.97	6.02
		512	10.50	12.02
		1024	5.25	24.03
Thinking Machines CM-2	Oct 91	8K	126.6	1.00
		16K	63.9	1.97
		32K	33.7	3.74
		64K	18.8	6.71
Thinking Machines CM-200	Oct 91	8K	76.9	1.64
		16K	39.2	3.22
		32K	20.7	6.10
		64K	10.9	11.58
Thinking Machines CM-5	Nov 92	16	42.4	2.98
		32	21.5	5.88
		64	10.9	11.62
		128	5.4	23.49
		256	2.7	46.84
		512	1.4	90.47
Thinking Machines CM-5E	Feb 94	32	11.5	11.0
		64	5.7	22.1
		128	3.0	42.1
PVM Sparcs (Ethernet)	Sep 93	16	1670.0	0.08
PVM RS6000-550 (Ethernet)	Sep 93	4	890.0	0.14
PVM Mixed-A (Ethernet)	Sep 93	34	494.0	0.26
PVM SGI 4D25 (Ethernet)	Sep 93	4	2536.4	0.05
Parasoft IBM (token)	Jan 94	9	589.0	0.2

Table 2a: (cont'd) Results of the Class A Embarrassingly Parallel (EP) Benchmark

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to C90/1
Cray C90	Dec 93	1	157.29*	1.00*
		4	39.62*	4.0*
		16	10.00*	15.7*
Cray T3D	Feb 94	32	41.77*	3.8*
		64	20.89*	7.5*
		128	10.44*	15.1*
		256	5.22*	30.1*
Fujitsu VPP500	Feb 94	1	193.47	na
		4	49.18	na
		16	12.39	na
IBM RS6000-590	Mar 94	1	272.25*	na
IBM SP-1	Mar 94	16	27.29*	na
		32	13.63*	na
		64	6.88*	na
Intel Paragon (OSF1.2)	Mar 94	64	41.74	na
		128	20.86	na
		256	10.47	na
		512	5.26	na
Intel Paragon (SunMos turbo)	Mar 94	64	21.18	na
		128	10.49	na
		256	5.41	na
nCUBE-2S	Mar 94	64	336.3	na
		128	168.2	na
		256	84.1	na
		512	42.1	na
		1024	21.0	na
Thinking Machines CM-5E	Feb 94	32	46.9	na
		64	23.6	na
		128	11.6	na

Table 2b: Results of the Class B Embarrassingly Parallel (EP) Benchmark (* indicates result based on reduced precision table lookup)

This code is a good test of both short and long distance highly structured communication. The Class B problem uses the same size grid but a greater number of inner loop iterations.

Results for this benchmark are shown in Table 3. Intel iPSC/860 and Paragon results are due to J. Patterson of BCS. CM-2 and CM-200 results are due to J. Richardson at TMC. KSR1 and KSR2 results are due to G. Montry of Southwest Software. VPP500 results are due to J.C.H. Wang of Fujitsu. Distributed workstation results are due to S. White of Emory University [12] using PVM 2.4 and Ethernet except where noted otherwise.

3.3 Conjugate Gradient (CG) Benchmark

In this benchmark, a conjugate gradient method is used to compute an approximation to the smallest eigenvalue of a large, sparse, symmetric positive definite matrix. This kernel is typical of unstructured grid computations in that it tests irregular long distance communication and employs sparse matrix vector multiplication.

An unfortunate inconsistency has developed in the specification of the Class A size CG benchmark. The original benchmark description (as written in RNR Technical Report RNR-91-002) specified 15 iterations, however subsequent publications (specifically [2]) specify 25 iterations. For historical consistency we continue to report timings for 15 iterations, and results we have received based on 25 iterations have been scaled by 15/25. (The benchmark time scales linearly with number of iterations.)

Results which have circumvented the sparse matrix-vector multiplication by retaining elements of the matrix construction, as described in Section 2.2, are marked by an asterisk.

The irregular communication requirement of this benchmark is evidently a challenge for all systems. Results are shown in Table 4. CM-2 results are due to J. Richardson of TMC. Intel iPSC/860 and nCUBE-2 results are by B. Hendrickson, R. Leland, and S. Plimpton of Sandia National Laboratory[7]. Paragon results are due to R. van de Geijn of U.T. Austin and John Lewis of BCS[8]. Cray EL and C90 results are due to RS6000-590 results are due to F. Gustavson of IBM. M. Zagha of Carnegie Mellon University. KSR1 and KSR2 results are due to S. Breit and J. Middlecoff of KSR. Distributed workstation results are due to S. White of Emory University [12] using PVM 2.4 and Ethernet except where noted otherwise.

3.4 3-D FFT PDE (FT) Benchmark

In this benchmark a 3-D partial differential equation is solved using FFTs. This kernel performs the essence of many "spectral" codes. It is a good test of long-distance communication performance.

The rules of the NAS Parallel Benchmarks specify that assembly-coded, library routines may be used to perform matrix multiplication and one-dimensional, two-dimensional or three-dimensional FFTs. Thus this benchmark is somewhat unique in that computational library routines may be legally employed.

Results are shown in Table 5. Intel iPSC/860 and Paragon results are due to E. Kushner of Intel. CM-2 and CM-200 results are due to J. Richardson of TMC. KSR1 and KSR2 results are due to N. Camp of KSR.

3.5 Integer Sort (IS) Benchmark

This benchmark tests a sorting operation that is important in "particle method" codes. This type of application is similar to "particle in cell" applications of physics, wherein particles are assigned to cells and may drift out. The sorting operation is used to reassign particles to the appropriate cells. This benchmark tests both integer computation speed and communication performance.

This problem is unique in that floating point arithmetic is not involved. Significant data communication, however, is required. Results are shown in Table 6. Intel iPSC/860 and Paragon results are due to to J. Baugh of Intel. CM-2, CM-200 and MasPar results use a library sorting routine. Cray Y-MP results are due to CRI. Cray C-90 and EL results are due to M. Zagha of Carnegie Mellon University using a radix sort optimized for interleaved memories [14]. KSR1 and KSR2 results are due to C. Nowacki of KSR. SP-1 results are due to B. Alpern and L. Carter of IBM.

Computer System	Date Received	No. Proc.	Time (sec)	Ratio to Y-MP/1
Cray Y-MP	Aug 92	1	22.22	1.00
		8	2.96	7.51
Cray EL	Aug 92	1	89.19	0.25
		4	27.94	0.80
		8	22.30	0.95
Cray C-90	Dec 93	1	8.15	2.7
		4	2.19	10.1
	Aug 92	16	0.96	23.14
Cray T3D	Feb 94	64	3.03	7.33
		128	1.56	14.15
		256	0.86	25.8
Fujitsu VPP500	Feb 94	4	1.78	12.5
		8	1.13	19.7
		16	0.79	28.1
IBM RS6000-590	Mar 94	1	41.78	0.53
IBM SP-1	Feb 94	8	34.83	0.6
		16	22.30	1.0
		32	15.04	1.5
		64	9.25	2.4
Intel iPSC/860	Aug 92	128	8.6	2.58
Intel Paragon (OSF1.2)	Mar 94	64	8.4	2.6
		128	4.5	4.9
		256	3.0	7.4
Intel Paragon (SunMos)	Feb 94	64	9.76	2.3
		128	5.10	4.4
		256	3.48	6.4
Kendall Square KSR1	Feb 94	32	19.7	1.1
		64	10.3	2.2
		128	5.6	4.0
Kendall Square KSR2	Feb 94	32	10.3	2.2
Kyoto/Matsushita ADENART	Feb 94	256	21.4	1.0
MasPar MP-1	Aug 92	16K	12.0	1.9
MasPar MP-2	Nov 92	16K	4.36	5.1
Meiko CS-1	Aug 92	16	42.8	0.5
nCUBE-2S	Mar 94	64	37.6	0.6
		128	19.2	1.2
		512	5.3	4.2
		1024	2.8	7.9
Thinking Machines CM-2	Dec 91	16K	45.8	0.5
		32K	26.0	0.9
		64K	14.1	1.6
Thinking Machines CM-200	Dec 91	16K	30.2	0.7
		32K	17.2	1.3
Thinking Machines CM-5	Aug 93	32	19.5	1.1
		64	10.9	2.0
		128	6.1	3.6
Thinking Machines CM-5E	Feb 94	32	3.9	5.7
		64	2.3	9.9
		128	1.3	16.6
PVM RS6000-550 (Ethernet)	Sep 93	4	293.0	0.1
PVM RS6000-560 (FDDI)	Sep 93	4	184.0	0.1
	Sep 93	8	110.4	0.2

Table 3a: Results of the Class A Multigrid (MG) Benchmark

Computer System	Date Received	No. Proc.	Time (sec)	Ratio to C90/1
Cray C90	Dec 93	1	37.77	1.0
		4	9.71	3.9
		16	3.97	9.5
Cray T3D	Mar 94	32	34.76	1.1
	Feb 94	128	7.57	5.1
		256	4.08	9.3
Fujitsu VPP500	Feb 94	4	8.65	4.4
		8	5.36	7.0
		16	3.81	9.9
IBM RS6000-590	Mar 94	1	184.92	0.2
IBM SP-1	Mar 94	16	97.86	0.4
		32	66.16	0.6
		64	44.02	0.9
Intel Paragon (OSF1.2)	Mar 94	64	39.8	0.9
		128	21.3	1.8
		256	13.7	2.8
Intel Paragon (SunMos)	Feb 94	64	43.02	0.9
		128	24.15	1.6
		256	16.74	2.3
Thinking Machines CM-5E	Feb 94	32	20.9	1.8
		64	11.3	3.3
		128	6.7	5.6

Table 3b: Results of the Class B Multigrid (MG) Benchmark

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
BBN TC2000	Dec 91	40	51.4	0.23
Cray Y-MP	Aug 92	1	11.92	1.00
		8	2.38	5.01
Cray EL	Sep 93	1	45.24	0.26
		4	14.29	0.83
		8	10.14	1.18
Cray C-90	Sep 93	1	3.55	3.36
		4	0.96	12.42
		16	0.34	35.06
Cray T3D	Sep 93	16	21.89	0.54
	Mar 94	32	10.50	1.14
		64	5.58	2.14
		128	3.29	3.62
		256	1.95	6.11
IBM RS6000-590	Mar 94	1	5.35*	2.23*
IBM SP-1	Feb 94	8	21.37	0.6
		16	12.82	0.9
		32	7.98	1.5
		64	4.72	2.5
Intel iPSC/860	Sep 93	128	7.0	1.71
Intel Paragon (OSF1.2)	Mar 94	64	4.10	2.9
		128	3.30	3.6
		256	2.83	4.2
Intel Paragon (SunMos)	Nov 93	64	12.6	1.0
Kendall Square KSR1	Feb 94	32	19.0	0.6
		64	13.4	0.9
Kendall Square KSR2	Feb 94	32	9.8	1.2
Kyoto/Matsushita ADENART	Feb 94	256	10.8	1.1
MasPar MP-1	Aug 92	4K	64.5	0.18
		16K	14.6	0.82
MasPar MP-2	Nov 92	16K	11.0	1.08
Meiko CS-1	Aug 92	16	67.5	0.18
nCUBE-2S	Mar 94	64	29.6	0.4
		128	16.9	0.7
		256	9.6	1.3
		512	6.2	1.9
		1024	4.1	2.9
Thinking Machines CM-2	Mar 92	8K	25.6	0.47
		16K	14.1	0.85
		32K	8.8	1.35
Thinking Machines CM-200	Mar 92	8K	15.0	0.79
Thinking Machines CM-5	Aug 93	32	20.7	0.58
		64	10.6	1.12
		128	6.2	1.92
Thinking Machines CM-5E	Feb 94	32	11.2	1.1
		64	5.8	2.1
		128	3.1	3.8
PVM RS6000-550 (Ethernet)	Sep 93	4	203.2	0.06
PVM RS6000-560 (FDDI)	Sep 93	4	81.5	0.15
Parasoft IBM (token)	Jan 94	9	277	0.04

Table 4a: Results of the Class A Conjugate Gradient (CG) Benchmark (* indicates result used matrix construction to circumvent sparse matrix-vector multiplication)

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to C90/1
Cray C90	Dec 93	1	122.90	1.00
		4	33.19	3.7
		16	10.61	11.6
Cray T3D	Mar 94	32	414.8	0.3
	Feb 94	128	118.2	1.0
		256	69.7	1.8
IBM RS6000-590	Mar 94	1	429.0*	0.3*
IBM SP-1	Mar 94	16	638.2	0.2
		32	362.9	0.3
		64	193.4	0.6
Intel Paragon (OSF1.2)	Mar 94	64	132.5	0.9
Thinking Machines CM-5E	Feb 94	32	449.0	0.3
		64	199.0	0.6
		128	92.0	1.3

Table 4b: Results of the Class B Conjugate Gradient (CG) Benchmark (* indicates result used matrix construction to circumvent sparse matrix-vector multiplication)

4 Simulated CFD Application Benchmarks

The three simulated CFD application benchmarks are intended to accurately represent the principal computational and data movement requirements of modern CFD applications.

The first of these is the called the lower-upper diagonal (LU) benchmark. It does not perform a LU factorization but instead employs a symmetric successive over-relaxation (SSOR) numerical scheme to solve a regular-sparse, block (5×5) lower and upper triangular system. This problem represents the computations associated with a newer class of implicit CFD algorithms, typified at NASA Ames by the code "INS3D-LU". This problem exhibits a somewhat limited amount of parallelism compared to the next two. Discussion of the serial algorithm underlying this benchmark may be found in [13]. Discussion of the parallel algorithms may be found in [3].

The second simulated CFD application is called the scalar pentadiagonal (SP) benchmark. In this benchmark, multiple independent systems of non-diagonally dominant, scalar pentadiagonal equations are solved. The third simulated CFD application is called the block tridiagonal (BT) benchmark. In this benchmark, multiple independent systems of non-diagonally dominant, block tridiagonal equations with a 5×5 block size are solved.

SP and BT are representative of computations associated with the implicit operators of CFD codes such as "ARC3D" at NASA Ames. SP and BT are similar in many respects, but there is a fundamental difference with respect to the communication to computation ratio. Discussion of the serial algorithm underlying this benchmark may be found in [10].

Performance figures for the three simulated CFD applications are shown in Tables 7, 8 and 9. Timings are cited as complete run times, in seconds, as with the other benchmarks. A complete solution of the LU benchmark requires 250 iterations. For the SP benchmark, 400 iterations are required. For the BT benchmark, 200 iterations are required.

For LU, credits are as follows: iPSC/860 and CM-2 results are due to S. Weeratunga, R. Fatoohi, E. Barszcz and V. Venkatakrishnan of NAS; CM-5 results are due to J. Richardson and D. Sandee of TMC; MP-1 and MP-2 results are due to J. McDonald of MasPar; Intel Paragon results are due to T. Phung of Intel; KSR1 and KSR2 results are due to S. Breit of KSR; SP-1 results are due to V. Naik of IBM; nCUBE-2S results are due to E. Schulman of nCUBE.

For SP, credits are as follows: CM-2 results employ a library scalar pentadiagonal solver; CM-5 results are due to J. Richardson and D. Sandee of TMC; iPSC/860 results are due to J. Patterson of BCS; Paragon results are due to T. Phung of Intel; MP-1 and MP-2 results are due to J. McDonald of MasPar; KSR1 and KSR2 results are due to S. Breit of KSR and G. Shah of Georgia Tech; SP-1 and RS6000-590 results are

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
Cray Y-MP	Aug 92	1	28.77*	1.00
		8	4.19*	6.87
Cray EL	May 93	1	105.1*	0.27
		4	27.9*	1.03
		8	18.5*	1.56
Cray C-90	Aug 92	1	10.28*	2.80
		4	2.58*	11.20
		16	0.91*	31.60
Cray T3D	Feb 94	32	6.42*	4.5
		64	3.28*	8.8
		128	1.67*	17.2
		256	0.86*	33.5
IBM RS6000-590	Feb 94	1	61.01*	0.5
IBM SP-1	Feb 94	8	43.68*	0.7
		16	22.86*	1.3
		32	12.08*	2.4
		64	6.46*	4.5
Intel iPSC/860	Dec 91	64	20.9*	1.37
	Apr 92	128	9.7*	2.96
Intel Paragon (OSF1.2)	Mar 94	64	9.1*	3.2
		128	4.9*	5.9
		256	3.6*	8.0
Intel Paragon (SunMos)	Mar 94	64	7.2*	4.0
		128	3.9*	7.4
		256	3.0*	9.7
Kendall Square KSR1	Feb 94	32	16.2*	1.8
		64	9.2*	3.1
Kendall Square KSR2	Feb 94	32	9.0*	3.2
Kyoto/Matsushita ADENART	Feb 94	256	72.7	0.4
MasPar MP-1	Aug 92	16K	18.3*	1.57
MasPar MP-2	Nov 92	16K	8.0*	3.60
Meiko CS-1	Aug 92	16	170.0*	0.17
nCUBE-2S	Mar 94	64	62.8*	0.5
		128	32.9*	0.9
		256	16.0*	1.8
		512	8.4*	3.4
		1024	4.1*	7.0
Thinking Machines CM-2	Dec 91	16K	37.0*	0.78
		32K	18.2*	1.58
		64K	11.4*	2.52
Thinking Machines CM-200	Dec 91	8K	45.6*	0.63
Thinking Machines CM-5	Aug 93	32	14.9*	1.93
		64	7.9*	3.64
		128	6.6*	4.36
Thinking Machines CM-5E	Feb 94	32	7.4*	3.9
		64	3.9*	7.4
		128	2.9*	9.9

Table 5a: Results of the Class A 3-D FFT PDE (FT) Benchmark (* indicates library result).

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to C90/1
Cray C90	Dec 93	1	127.44*	1.00
		2	63.74*	2.0
		16	8.43*	15.1
Cray T3D	Feb 94	256	11.46*	11.1
IBM RS6000-590	Mar 94	1	856.3*	0.1
IBM SP-1	Mar 94	16	286.5*	0.4
		32	143.2*	0.9
		64	74.5*	1.7
Intel Paragon (OSF1.2)	Mar 94	128	56.5*	2.3
		256	30.6*	4.2
Intel Paragon (SunMos)	Feb 94	256	25.1*	5.1
Thinking Machines CM-5E	Feb 94	32	89.0*	1.4
		64	46.0*	2.8
		128	34.0*	3.7

Table 5b: Results of the Class B 3-D FFT PDE (FT) Benchmark (* indicates library result).

due to V. Naik of IBM; VPP500 results are due to S. Gavali of Fujitsu; nCUBE-2S results are due to E. Schulman of nCUBE.

For BT, credits are as follows: CM-2 and CM-200 results employ a library block tridiagonal solver; CM-5 results are due to J. Richardson and D. Sandee of TMC; iPSC/860 results are due to J. Patterson of BCS; Paragon results are due to T. Phung of Intel; MP-1 and MP-2 results are due to J. McDonald of MasPar; KSR1 and KSR2 results are due to S. Breit of KSR; SP-1 results are due to V. Naik of IBM; VPP500 results are due to H. Lai of Fujitsu; nCUBE-2S results are due to E. Schulman of nCUBE.

5 Sustained Performance Per Dollar

One aspect of the relative performance of these systems has not been addressed so far, namely the differences in price between these systems. One should not be too surprised that the Cray C-90 system, for example, exhibits superior performance rates on these benchmarks, since its current list price is much greater than that of any other system tested.

One way to compensate for these price differences is to compute sustained performance per million dollars, i.e. the performance ratio figures shown in Tables 2 through 9 divided by the list price in millions. Some figures of this type are shown in Table 10 for two of the benchmarks (the Class B size MG and SP benchmarks) for the most recent of the systems tested. The table includes the list price of the minimal system (in terms of memory per node, disk space, etc.) required to run the full Class B size NPB as implemented by the vendor. These prices were provided by the vendors and include any associated software costs (i.e. operating system, compilers, scientific libraries as required, etc.) but do not include maintenance. List prices for the various systems are as follows: Cray C90 with 16 processors, 256 Mwords is $30.90 million (Oct 93); Cray EL98 with 1 GB of memory and 12 GB disk is $1.11 million (Oct 93); Cray T3D with 256 nodes and 16 MB/node is $9.25 million (Mar 94); Fujitsu VPP500 with 16 nodes and 256 MB/node is $17.0 million (Mar 94); IBM SP-1 with 64 nodes, 64 MB/node, 64 GB disk is $2.66 million (Oct 93); IBM RS6000-590 with 1 GB memory is $0.25 million (Mar 94); Intel Paragon with 256 nodes, 32 MB/node is $7.49 million (Mar 94); Kendall Square KSR1 with 128 nodes, 32 MB/node, 25 GB disk is $1.7 million (Mar 94); Kendall Square KSR2 with 32 nodes, 32 MB/node, 25 GB disk is $1.43 million (Mar 94); MasPar MP-2 with 16K processors, 1 GB memory, DEC front end is $ 1.61 million (Oct 93); nCUBE-2S with 1024 nodes, 4 MB/node is $4.0 million (Mar 94); TMC CM-5E with 128 nodes (with vector units, 40 MHz SuperSPARC and 40 MHz data router), 32 MB/node is $4.00 million (Mar 94); Note that some vendor standard configurations may include substantially more hardware than required for the benchmarks (for example, the IBM SP-1). Finally, be aware that list prices are similar to peak performance in that they are guaranteed not to be exceeded.

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
Cray Y-MP	Aug 92	1	11.46	1.00
		8	1.85	6.19
Cray EL	Sep 93	1	43.76	0.26
		4	12.99	0.88
		8	8.45	1.35
Cray C-90	Sep 93	1	3.33	3.44
		4	0.85	13.46
		16	0.27	42.38
Cray T3D	Feb 94	32	7.04	1.6
		64	3.42	3.4
		128	1.75	6.5
		256	0.92	12.5
IBM RS6000-590	Feb 94	1	21.73	0.5
IBM SP-1	Feb 94	8	16.81	0.7
		16	8.85	1.3
		32	5.04	2.3
		64	3.06	3.7
Intel iPSC/860	May 92	32	25.7	0.45
		64	17.3	0.66
		128	13.6	0.84
Intel Paragon (OSF1.2)	Mar 94	32	7.81	1.5
		64	4.34	2.6
		128	2.41	4.8
Intel Paragon (SunMos)	Mar 94	32	5.48	2.1
		64	3.77	3.0
Kendall Square KSR1	Feb 94	32	10.8	1.1
		64	6.6	1.7
Kendall Square KSR2	Feb 94	32	7.0	1.6
Kyoto/Matsushita ADENART	Feb 94	256	46.6	0.3
MasPar MP-1	Jan 93	16K	11.5*	1.00
MasPar MP-2	Jan 93	16K	7.7*	1.49
Meiko CS-1	Aug 92	16	62.7	0.18
nCUBE-2S	Mar 94	64	23.2	0.5
		128	12.0	1.0
		256	6.1	1.9
		512	3.2	3.6
		1024	1.7	6.8
Thinking Machines CM-2	Dec 91	16K	35.8*	0.32
		32K	21.0*	0.55
		64K	14.9*	0.77
Thinking Machines CM-200	Dec 91	64K	5.7*	2.01
Thinking Machines CM-5	Aug 93	32	43.1	0.27
		64	24.2	0.47
		128	12.0	0.96
Thinking Machines CM-5E	Feb 94	32	6.3	1.8
		64	3.1	3.7
		128	1.66	6.9

Table 6a: Results of the Class A Integer Sort (IS) Benchmark (* indicates library result).

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to C90/1
Cray C90	Dec 93	1	12.92	1.00
		4	3.30	3.9
		16	0.98	13.7
Cray T3D	Mar 94	32	29.90	0.4
	Feb 94	128	7.70	1.7
		256	3.82	3.4
IBM RS6000-590	Mar 94	1	91.6	0.1
IBM SP-1	Mar 94	16	37.3	0.3
		32	20.1	0.6
		64	11.2	1.2
Intel Paragon (OSF1.2)	Mar 94	64	17.33	0.7
		128	9.52	1.4
		256	5.94	2.2
		512	4.69	2.8
Intel Paragon (SunMos)	Mar 94	64	11.98	1.1
		128	7.22	1.8
nCUBE-2S	Mar 94	128	47.5	0.3
		512	12.5	1.0
		1024	6.5	2.0
Thinking Machines CM-5E	Feb 94	32	32.0	0.4
		64	16.4	0.8
		128	8.4	1.5

Table 6b: Results of the Class B Integer Sort (IS) Benchmark

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
BBN TC2000	Dec 91	62	3032.0	0.11
Cray Y-MP	Aug 92	1	333.5	1.00
		8	49.5	6.74
Cray EL	Aug 92	1	1449.0	0.23
		4	522.3	0.64
		8	351.6	0.95
Cray C-90	Aug 92	1	157.6	2.12
		4	43.9	7.59
		16	17.6	18.93
Cray T3D	Feb 94	32	178.0	1.9
		64	96.8	3.4
		128	51.2	6.5
		256	28.1	11.9
IBM RS6000-590	Mar 94	1	645.2	0.5
IBM SP-1	Feb 94	8	291.4	1.1
		16	172.9	1.9
		32	101.8	3.3
		64	63.2	5.3
Intel iPSC/860	Mar 91	64	690.8	0.48
		128	442.5	0.75
Intel Paragon (OSF1.2)	Mar 94	64	364.0	0.9
		128	231.0	1.4
		256	152.0	2.2
Intel Paragon (SunMos)	Mar 94	64	344.0	1.0
		128	230.0	1.5
		256	158.0	2.1
Kendall Square KSR1	Feb 94	32	341	1.0
		64	199	1.7
		128	155	2.2
Kendall Square KSR2	Feb 94	32	172	1.9
Kyoto/Matsushita ADENART	Feb 94	256	327.5	1.0
MasPar MP-1	Aug 92	4K	1580.0	0.2
MasPar MP-2	Nov 92	4K	463.5	0.7
Meiko CS-1	Aug 92	16	2937.0	0.1
nCUBE-2S	Mar 94	64	1322.0	0.3
		128	712.5	0.5
		256	389.1	0.9
		512	226.1	1.5
		1024	134.1	2.5
Thinking Machines CM-2	Mar 91	8K	1307.0	0.26
		16K	850.0	0.39
		32K	546.0	0.61
Thinking Machines CM-5	Aug 93	32	418.0	0.80
		64	272.0	1.23
		128	171.0	1.95
Thinking Machines CM-5E	Feb 94	32	152.0	2.2
		64	97.0	3.4
		128	65.0	5.1

Table 7a: Results for the Class A LU Simulated CFD Application

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to C90/1
Cray C90	Dec 93	1	648.5	1.00
		4	166.1	3.9
		16	51.6	12.6
Cray T3D	Mar 94	32	738.0	0.9
	Feb 94	128	211.6	3.1
		256	119.5	5.4
IBM RS6000-590	Mar 94	1	2694.6	0.2
IBM SP-1	Feb 94	16	604.8	1.1
		32	348.1	1.9
		64	207.5	3.1
Intel Paragon (OSF1.2)	Mar 94	102	975.0	0.7
		204	524.0	1.2
		384	380.0	1.7
Intel Paragon (SunMos)	Mar 94	102	899.0	0.7
		204	510.0	1.3
		384	387.0	1.7
Thinking Machines CM-5E	Feb 94	32	595.0	1.1
		64	367.0	1.8
		128	318.0	2.0

Table 7b: Results for the Class B LU Simulated CFD Application

References

[1] D. H. Bailey, E. Barszcz, J. T. Barton, D. S. Browning, R. L. Carter, L. Dagum, R. A. Fatoohi, P. O. Frederickson, T. A. Lasinski, R. S. Schreiber, H. D. Simon, V. Venkatakrishnan, and S. K. Weeratunga, "The NAS Parallel Benchmarks", *Intl. Journal of Supercomputer Applications*, v. 5, no. 3 (Fall 1991), pp. 63 – 73.

[2] D. Bailey, J. Barton, T. Lasinski, and H. Simon, eds., "The NAS Parallel Benchmarks", NASA Technical Memorandum 103863, Ames Research Center, Moffett Field, CA 94035-1000, July 1993.

[3] E. Barszcz, R. Fatoohi, V. Venkatakrishnan, and S. Weeratunga, "Solution of Regular Sparse Triangular Linear Systems on Vector and Distributed Memory Multiprocessors", Tech Report RNR-93-07, NASA Ames Research Center, Moffett Field, CA 94035, April 1993.

[4] G. Bhanot, K. Jordan, J. Kennedy, J. Richardson, D. Sandee and M. Zagha, "Implementing the NAS Parallel Benchmarks on the CM-2 and CM200 Supercomputers", Thinking Machines Corp, Cambridge, MA 02142.

[5] S. Breit, W. Celmaster, W. Coney, R. Foster, B. Gaiman, G. Montry and C. Selvidge, "The Role of Computational Balance in the Implementation of the NAS parallel Benchmarks on the BBN TC2000 Computer", submitted to *Concurrency*, April 1991.

[6] M. Fillo, *Architectural Support for Scientific Applications on Multicomputers*, Hartung Gorre Verlag, Series in Microelectronics, Volume 27, Konstanz, Germany, 1993.

[7] B. Hendrickson, R. Leland, and S. Plimpton, "An Efficient Parallel Algorithm for Matrix-Vector Multiplication", Sandia Report SAND92-2765, Sandia National Lab, Albuquerque, NM 87185, March 1993.

[8] J. Lewis, and R. van de Geijn, "Distributed Memory Matrix-vector Multiplication and Conjugate Gradient Algorithms", in Proceedings Supercomputing '93, Portland, OR, Nov. 15-19, 1993.

[9] V. K. Naik, "Performance Issues in Implementing NAS Parallel Benchmark Applications on IBM SP-1", Research Report, T.J. Watson Research Center, IBM, (in preparation) 1993.

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
BBN TC2000	Dec 91	112	880.0	0.54
Cray Y-MP	Aug 92	1	471.5	1.00
		8	64.6	7.30
Cray EL	Aug 92	1	2025.7	0.23
		4	601.9	0.78
		8	488.4	0.97
Cray C-90	Aug 92	1	184.70	2.55
		4	49.74	9.48
		16	13.06	36.10
Cray T3D	Feb 94	32	190.8	2.5
		64	97.5	4.8
		128	49.7	9.5
		256	25.6	18.4
Fujitsu VPP500	Feb 94	1	244.9	1.9
		2	156.0	3.0
		4	80.3	5.9
		8	42.0	11.2
		16	28.9	16.3
IBM RS6000-590	Mar 94	1	993.1	0.5
IBM SP-1	Feb 94	8	441.6	1.1
		16	268.7	1.8
		32	165.0	2.9
		64	100.4	4.7
Intel iPSC/860	Aug 92	64	667.3	0.71
		128	449.5	1.05
Intel Paragon (OSF1.2)	Mar 94	64	257.0	1.8
		100	177.0	2.7
		256	97.0	4.9
		324	89.0	5.3
Kendall Square KSR1	Feb 94	32	418	1.1
		64	257	1.8
		128	160	2.9
Kendall Square KSR2	Feb 94	32	221	2.1
Kyoto/Matsushita ADENART	Feb 94	256	209.9	2.3
MasPar MP-1	Aug 92	4K	1772	0.27
MasPar MP-2	Nov 92	4K	615	0.77
Meiko CS-1	Aug 92	16	2975	0.16
nCUBE-2S	Mar 94	64	1243.2	0.4
		128	717.4	0.7
		256	528.5	0.9
		512	281.5	1.7
		1024	235.8	2.0
Thinking Machines CM-2	Dec 91	16K	1444.0*	0.33
		32K	917.0*	0.51
		64K	640.0*	0.74
Thinking Machines CM-5	May 93	32	289.0	1.63
		64	170.0	2.77
		128	119.0	3.96
Thinking Machines CM-5E	Feb 94	32	169.0	2.8
		64	104.0	4.5
		128	61.0	7.7

Table 8a: Results for the Class A SP Simulated CFD Application (* indicates library result).

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to C90/1
Cray C90	Dec 93	1	713.1	1.00
		4	203.1	3.5
		16	80.4	8.9
Cray T3D	Mar 94	32	849.8	0.8
	Feb 94	128	224.7	3.2
		256	124.2	5.7
IBM RS6000-590	Mar 94	1	4047.2	0.2
IBM SP-1	Feb 94	16	941.2	0.8
		32	522.4	1.4
		64	302.3	2.5
Intel Paragon (OSF1.2)	Mar 94	64	960.0	0.7
		100	649.0	1.1
		256	301.0	2.4
		324	262.0	2.7
		400	246.0	2.9
		484	209.0	3.4
Thinking Machines CM-5E	Feb 94	32	1014.0	0.7
		64	595.0	1.2
		128	320.0	2.2

Table 8b: Results for the Class B SP Simulated CFD Application.

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
BBN TC2000	Dec 91	112	1378.0	0.58
Cray Y-MP	Aug 92	1	792.4	1.00
		8	114.0	6.95
Cray EL	May 93	1	3832.8	0.21
		4	1090.2	0.73
		8	764.1	1.04
Cray C-90	Aug 92	1	356.9	2.22
		4	96.1	8.25
		16	28.4	27.91
Cray T3D	Feb 94	32	263.9	3.0
		64	132.7	6.0
		128	66.2	12.0
		256	33.8	23.4
Fujitsu VPP500	Feb 94	2	166.7	4.8
		4	88.8	8.9
		8	47.8	16.6
		16	29.7	26.7
IBM RS6000-590	Feb 94	1	1249.4	0.6
IBM SP-1	Feb 94	8	534.6	1.5
		16	297.6	2.7
		32	167.3	4.7
		64	95.2	8.3

Table 9a: Results for the Class A BT Simulated CFD Application (* indicates library result) (cont'd).

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to Y-MP/1
Intel iPSC/860	Aug 92	64	714.7	1.11
		128	414.3	1.91
Intel Paragon (OSF1.2)	Mar 94	64	235.0	3.4
		128	129.0	6.1
		256	83.0	9.5
		512	63.0	12.5
Intel Paragon (SunMos)	Nov 93	64	224.0	3.5
	Mar 94	128	113.0	7.0
Kendall Square KSR1	Feb 94	32	457	1.7
		64	256	3.1
		128	145	5.5
Kendall Square KSR2	Feb 94	32	225	3.5
Kyoto/Matsushita ADENART	Feb 94	256	314.1	2.5
MasPar MP-1	Aug 92	4K	2396.0	0.33
MasPar MP-2	Nov 92	4K	789.0	1.00
Meiko CS-1	Aug 92	16	2984.0	0.27
nCUBE-2S	Mar 94	64	1243.2	0.6
		128	644.7	1.2
		256	388.2	2.0
		512	211.4	3.8
		1024	151.7	5.2
Thinking Machines CM-2	Dec 91	16K	1118.0*	0.71
		32K	634.0*	1.25
		64K	370.0*	2.14
Thinking Machines CM-200	Dec 91	16K	832.0*	0.95
		32K	601.0*	1.32
Thinking Machines CM-5	May 93	32	284.0	2.79
		64	175.0	4.50
		128	119.0	6.66
Thinking Machines CM-5E	Feb 94	32	146.0	5.4
		64	84.0	9.4
		128	48.0	16.5

Table 9a: (cont'd) Results for the Class A BT Simulated CFD Application (* indicates library result).

Computer System	Date Received	No. Proc.	Time (sec.)	Ratio to C90/1
Cray C90	Dec 93	1	1261.4	1.00
		4	324.9	3.9
		16	96.4	13.1
Cray T3D	Mar 94	32	1080.1	1.2
	Feb 94	128	306.9	4.1
		256	166.4	7.6
IBM RS6000-590	Mar 94	1	5242.4	0.2
IBM SP-1	Feb 94	16	1179.1	1.1
		32	607.5	2.1
		64	326.9	3.9
Fujitsu VPP500	Feb 94	17	86.7	14.6
Intel Paragon (OSF1.2)	Mar 94	102	633.0	2.0
		204	359.0	3.5
		306	257.0	4.9
		408	226.0	5.6
		510	196.0	6.4
Intel Paragon (SunMos)	Mar 94	102	598.0	2.1
		204	324.0	3.9
		306	215.0	5.9
Thinking Machines CM-5E	Feb 94	32	806.0	1.6
		64	464.0	2.7
		128	253.0	5.0

Table 9b: Results for the Class B BT Simulated CFD Application

B'mark	Computer System	No. Proc.	Ratio to C90/1	Nominal cost ($)	Perf. per million $
MG-B	Cray C-90	16	9.5	30.90M	0.31
	Cray T3D	256	9.3	9.25M	1.01
	Fujitsu VPP500	16	9.9	17.00M	0.58
	IBM SP-1	64	0.9	2.66M	0.34
	IBM RS6000-590	1	0.2	0.25M	0.82
	Intel Paragon (OSF1.2)	256	2.8	7.49M	0.37
	Thinking Machines CM-5E	128	5.6	4.00M	1.40
SP-B	Cray C-90	16	8.9	30.90M	0.29
	Cray T3D	256	5.7	9.25M	0.62
	IBM SP-1	64	2.5	2.66M	0.94
	IBM RS6000-590	1	0.18	0.25M	0.70
	Intel Paragon	256	2.4	7.49M	0.32
	Thinking Machines CM-5E	128	2.2	4.00M	0.55

Table 10: Approximate Sustained Performance Per Dollar on Two Class B Benchmarks

[10] T. H. Pulliam. "Efficient Solution Methods for the Navier-Stokes Equations", Lecture Notes for The Von Karman Institute for Fluid Dynamics Lecture Series, Jan. 20 - 24, 1986.

[11] F. Sukup and J. Fritscher, "Efficiency Evaluation of Some Parallelization Tools on a Workstation Cluster Using the NAS Parallel Benchmarks", Computing Center, Vienna University of Technology. Vienna, Austria.

[12] S. White, "NAS Benchmark on Virtual Parallel Machines", Master's thesis, Emory University, 1993.

[13] S. Yoon, D. Kwak, and L. Chang, "LU-SGS Implicit Algorithm for Implicit Three Dimensional Navier-Stokes Equations with Source Term", AIAA Paper 89-1964-CP, American Institute of Aeronautics and Astronautics, Washington, D.C., 1989.

[14] M. Zagha and G.E. Blelloch, "Radix Sort for Vector Multiprocessors", Proceedings of Supercomputing '91, pp. 712-721, New York, NY, Nov. 1991.